

13/17

Application No. 09/715,981

Atty Docket No. PUMA 1019-1 – SF/0029.01

REMARKS

Claims 1-29 are pending in this application, and stand rejected under 35 USC 102 as being anticipated by Boothby et al. (USP 6,141,664).

Applicant appreciates the Examiner's consideration of whether this Office Action should be final or not, while disagreeing with the Examiner's deletion of section 5, which explained that the positions taken by the Applicant did not need to be answered because they were moot, in light of new grounds of rejection. During a brief interview, Applicant pointed out that the Examiner is making new arguments and has not responded to the positions previously taken. In the remarks that follow, this theme is much further developed than in a brief interview. We did not expect the Examiner to respond by deleting section 5.

Preliminarily, further discussion of the concept of inheritance, as opposed to ordinary synchronization, may help the Examiner appreciate the differences between Boothby and the present claims. From page 17 of the application,

The Infomanager 301 maintains and controls a number of datasets including, for example, a parent dataset 303 and a child dataset 305. The child dataset 305 can inherit information from the parent dataset 303. ... In general, information may flow from the parent dataset 303 into the child dataset 10 305 via inheritance, but the child dataset 305 needs not alter the parent dataset 303. This generally or largely one-way flow of information, controlled by the Infomanager 301, is shown schematically in FIG. 3A by a uni-directional solid arrow (e.g., "flow arrow") that couples the parent dataset 303 and the child dataset 305.

The example of FIG. 3B, presenting various levels and features of inheritance, depicts propagating parts of a 1999 Major League Baseball through several levels into an individual's schedule. Obviously, an individual can annotate the schedule for a particular game, but cannot effectively reschedule a Major League event.

"Inheritance-aware" handling of modifications to inherited items is much different than simply mapping between data items among two co-equal datasets, either of which can be altered with synchronization and propagation of changes to the other.

Inheritance-aware synchronization between a so-called child and its so-called alter-ego is discussed in the application, section VII, pp. 42 *et seq.*, and effectively contrasted with parent-child inheritance. With this distinction between parent-to-child

14/17

Application No. 09/715,981**Atty Docket No. PUMA 1019-1 – SF/0029.01**

inheritance and child-to-alter ego synchronization in mind, we address the rejections.

Regarding claim 1, Applicant previously established that passages of Boothby at col. 5, lines 5-37 and col. 6, lines 35-39, do not teach the receiving, placing or when processing steps. The remarks on page 10 of the prior response were not answered by the Examiner, except by citing additional passages from Boothby. The Examiner has either acquiesced in Applicant's position or not yet answered regarding the col. 5-6 passages. As stated in the now-deleted section 5, new grounds of rejection are now relied upon, in the sense that reliance on the old passages was not defended and new passages are now relied upon.

The Examiner now adds col. 8, lines 1-51 and col. 19, lines 25-63 as cited passages. The new passage col. 8 passage refers to an extended index array, including pointers, that appears to be temporarily constructed for efficient processing. The new col. 19 passage discusses mapping between sections or databases used by two applications, such as scheduling applications.

Neither the previously nor the currently cited passages includes any parent-child relationship between datasets or any concept of inheritance. The A-B and B-A maps to which the Examiner refers and the related discussion make it clear that data is synchronized between the A_Database and B_Database, without any inheritance relationship or any inheritance awareness. Boothby certainly does not refer to any data as being inherited – the word "inherit" is not found in Boothby. Therefore, Boothby does not disclose "receiving a first user input ... selecting a first data item from ... for inheritance." Mapping (A-B and B-A) two databases for synchronization purposes is not the same as and does not anticipate selecting data for inheritance.

Temporarily constructing an extended index array does not anticipate "placing a first pointer" in the first dataset pointing to an inherited record in the second dataset, because Boothby does not teach inheritance. Moreover, the extended index array is not either the first dataset or the second dataset, in Boothby's words.

Therefore, claim 1 should be allowable over Boothby, because Boothby does not teach any sort of inheritance, much less the claimed method of implementing

15/17

Application No. 09/715,981**Atty Docket No. PUMA 1019-1 – SF/0029.01**

inheritance.

Claim 14 is the second independent claim that the Examiner addresses.

Applicant previously established that passages of Boothby at col. 5, lines 21-38 and col. 8, lines 3-33, do not teach the processing steps. The remarks on page 11 of the prior response were not answered by the Examiner, except by citing additional passages from Boothby. The Examiner has either acquiesced in Applicant's position or not yet answered regarding the col. 5 & 8 passages.

The Examiner now adds a few lines, col. 6, lines 4-16, as a cited passage. This passage refers to field types of intermediate, translation data structures. The Examiner also adds from col. 19, lines 40-42, which lines are discussed above.

Again, neither the previously nor the currently cited passages includes any parent-child relationship between datasets or any concept of inheritance. As applied to claim 14, there is no distinction in Boothby between native and inherited data. All of Boothby's data would likely be classified as native, according to the present disclosure. Again, Boothby does not teach any sort of inheritance, much less the claimed method of implementing inheritance side-by-side with native data.

Therefore, claim 14 should be allowable over Boothby.

Claim 17 is the third independent claim that the Examiner addresses.

Applicant previously explained that passages of Boothby at col. 6, lines 35-39, col. 8, lines 3-33 and col. 10, line 10 to col. 11, line 20, do not teach the claimed system. The remarks on page 11 of the prior response were not answered by the Examiner, but the Examiner gave new reasoning on pp. 15-16 of this Office Action, rather than just citing passages. Applicant now responds to the new reasoning and argument.

The claimed system inherits data at both the record level and the dataset level. It applies a particular processing, by virtue of the wherein limitation. Boothby does not teach a choice between operating at a record level or a dataset level. Boothby, once again, does not teach inheritance, much less implementation of inheritance by pointers in the first dataset or a choice between inheriting at a record level or a

16/17

Application No. 09/715,981**Atty Docket No. PUMA 1019-1 – SF/0029.01**

dataset level.

Therefore, claim 17 should be allowable over Boothby.

Claim 23 is the fourth independent claim that the Examiner addresses.

Applicant previously explained that passages of Boothby at col. 5, lines 5-37 and col. 6, lines 35-67, do not teach the claimed system. The remarks on page 13 of the prior response were not answered by the Examiner, but the Examiner cited a new passage (col. 19, lines 27-28) and argued that the “address section is the ancestor of Business address and Personal Address”. Applicant now responds to the new reasoning and argument.

Boothby col. 19, lines 27-30, read:

Each of these databases are known as sections. Each of these sections contain different data an must be synchronized with their corresponding sections in other Applications.

This passage refutes the notion that one section is a parent and the other a child. In the terms of this disclosure, the two databases described by Boothby in col. 19 are alter-egos of one another, with no parent-child relationship. There is no distinction in Boothby between the relationship of a child dataset to an ancestor dataset from which a data item is inherited and an alter-ego dataset, with which data is synchronized. Boothby does not teach inheritance, as distinct from synchronization.

Therefore, claim 23 should be allowable over Boothby.

Dependent claims should be allowable for at least the same reasons as the independent claims. Additional reasons may be expressed, once applicants have succeeded in conveying to the Examiner begins the expressed distinction between native and inherited data, between synchronization and inheritance. For instance, filtering in claim 2 is not discussed in Boothby. Local editing to annotate an inherited data item in a way that will not be reported to the parent dataset, in claims 3, 12, and 20, is not discussed in Boothby. Duplicate detection from paths of inheritance in claim 21 is not discussed in Boothby. See generally the remarks on pages 14-17 of the prior response.

Application No. 09/715,981

Atty Docket No. PUMA 1019-1 – SF/0029.01

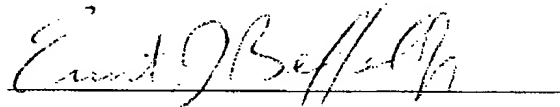
CONCLUSION

Applicant respectfully submits that the claims, as stated herein, are in condition for allowance and solicits acceptance of the claims, in light of these remarks. If the Examiner disagrees and sees amendments that might facilitate allowance of the claims, a call to the undersigned would be appreciated.

An interview to discuss how this application discloses inheritance might be useful to the Examiner, as the disclosure is long.

Should any questions arise, the undersigned can ordinarily be reached at his office at 650-712-0340 from 8:30 to 5:30 PST, M-F and can be reached at his cell phone 415-902-6112 most other times.

Respectfully submitted,



Ernest J. Beffel, Jr., Reg. No. 43,489

Dated: 13 May 2004

HAYNES BEFFEL & WOLFELD LLP
P.O. Box 366
Half Moon Bay, CA 94019
(650) 712-0340 phone
(650) 712-0263 fax